



**SLOVENSKI STANDARD**  
**oSIST prEN IEC 62541-10:2024**  
**01-marec-2024**

---

**Enotna arhitektura OPC - 10. del: Programi**

OPC Unified Architecture - Part 10: Programs

OPC Unified Architecture - Teil 10: Programme

Architecture unifiée OPC - Partie 10: Programmes

**Ta slovenski standard je istoveten z: prEN IEC 62541-10:2024**

---

**ICS:**

25.040.40	Merjenje in krmiljenje industrijskih postopkov	Industrial process measurement and control
35.240.50	Uporabniške rešitve IT v industriji	IT applications in industry

**oSIST prEN IEC 62541-10:2024**                      **en,fr,de**





PROJECT NUMBER: <b>IEC 62541-10 ED4</b>	
DATE OF CIRCULATION: <b>2024-01-26</b>	CLOSING DATE FOR VOTING: <b>2024-04-19</b>
SUPERSEDES DOCUMENTS: <b>65E/982/RR</b>	

IEC SC 65E : DEVICES AND INTEGRATION IN ENTERPRISE SYSTEMS	
SECRETARIAT: United States of America	SECRETARY: Mr Donald (Bob) Lattimer
OF INTEREST TO THE FOLLOWING COMMITTEES:	PROPOSED HORIZONTAL STANDARD: <input type="checkbox"/> Other TC/SCs are requested to indicate their interest, if any, in this CDV to the secretary.
FUNCTIONS CONCERNED: <input type="checkbox"/> EMC <input type="checkbox"/> ENVIRONMENT <input type="checkbox"/> QUALITY ASSURANCE <input type="checkbox"/> SAFETY	
<input checked="" type="checkbox"/> SUBMITTED FOR CENELEC PARALLEL VOTING <b>Attention IEC-CENELEC parallel voting</b> The attention of IEC National Committees, members of CENELEC, is drawn to the fact that this Committee Draft for Vote (CDV) is submitted for parallel voting. The CENELEC members are invited to vote through the CENELEC online voting system.	<input type="checkbox"/> NOT SUBMITTED FOR CENELEC PARALLEL VOTING

oSIST prEN IEC 62541-10:2024

<https://standards.iteh.ai> This document is still under study and subject to change. It should not be used for reference purposes. [ren-iec-62541-10-2024](https://standards.iteh.ai)

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Recipients of this document are invited to submit, with their comments, notification of any relevant "In Some Countries" clauses to be included should this proposal proceed. Recipients are reminded that the CDV stage is the final stage for submitting ISC clauses. (SEE [AC/22/2007](#) OR [NEW GUIDANCE DOC](#)).

TITLE:

**OPC Unified Architecture - Part 10: Programs**

PROPOSED STABILITY DATE: 2026

NOTE FROM TC/SC OFFICERS:

**Copyright © 2023 International Electrotechnical Commission, IEC.** All rights reserved. It is permitted to download this electronic file, to make a copy and to print out the content for the sole purpose of preparing National Committee positions. You may not copy or "mirror" the file or printed version of the document, or any part of it, for any other purpose without permission in writing from IEC.

## CONTENTS

1			
2			
3	1	Scope .....	1
4	2	Normative references .....	1
5	3	Terms, definitions and conventions .....	1
6	3.1	Terms and definitions .....	1
7	3.2	Abbreviations .....	2
8	4	Concepts .....	2
9	4.1	General .....	2
10	4.2	Programs .....	3
11	4.2.1	Overview .....	3
12	4.2.2	Security considerations .....	4
13	4.2.3	Program Finite State Machine .....	4
14	4.2.4	Program states .....	5
15	4.2.5	State transitions .....	6
16	4.2.6	Program state transition stimuli .....	6
17	4.2.7	Program Control Methods .....	6
18	4.2.8	Program state transition effects .....	7
19	4.2.9	Program result data .....	7
20	4.2.10	Program lifetime .....	8
21	5	Model .....	8
22	5.1	General .....	8
23	5.2	ProgramStateMachineType .....	9
24	5.2.1	Overview .....	9
25	5.2.2	ProgramStateMachineType Properties .....	11
26	5.2.3	ProgramStateMachineType components .....	12
27	5.2.4	ProgramStateMachineType causes (Methods) .....	15
28	5.2.5	ProgramStateMachineType effects (Events) .....	16
29	5.2.6	AuditProgramTransitionEventType .....	18
30	5.2.7	FinalResultData .....	18
31	5.2.8	ProgramDiagnostic2 DataType .....	18
32	5.2.9	ProgramDiagnostic2Type VariableType .....	19
33		Annex A (informative) Program example .....	21
34	A.1	Overview .....	21
35	A.2	DomainDownload Program .....	21
36	A.2.1	General .....	21
37	A.2.2	DomainDownload states .....	21
38	A.2.3	DomainDownload transitions .....	22
39	A.2.4	DomainDownload Methods .....	23
40	A.2.5	DomainDownload Events .....	24
41	A.2.6	DomainDownload model .....	24
42			

**FIGURES**

43		
44		
45	Figure 1 – Automation facility control .....	3
46	Figure 2 – Program illustration .....	4
47	Figure 3 – Program states and transitions .....	5
48	Figure 4 – Program Type .....	9
49	Figure 5 – Program FSM References .....	12
50	Figure 6 – ProgramStateMachineType causes and effects .....	15
51	Figure A.1 – Program example .....	21
52	Figure A.2 – DomainDownload state diagram .....	22
53	Figure A.3 – DomainDownloadType partial state model .....	28
54	Figure A.4 – Ready To Running model .....	30
55	Figure A.5 – Opening To Sending To Closing model .....	32
56	Figure A.6 – Running To Suspended model .....	33
57	Figure A.7 – Suspended To Running model .....	34
58	Figure A.8 – Running To Halted – Aborted model .....	35
59	Figure A.9 – Suspended To Aborted model .....	35
60	Figure A.10 – Running To Completed model .....	36
61	Figure A.11 – Sequence of operations .....	37
62		

iTeh Standards  
(<https://standards.iteh.ai>)  
Document Preview

[oSIST prEN IEC 62541-10:2024](https://standards.iteh.ai/catalog/standards/sist/1b349aca-5812-41e3-88fa-939a42f9efc4/osist-pren-iec-62541-10-2024)

<https://standards.iteh.ai/catalog/standards/sist/1b349aca-5812-41e3-88fa-939a42f9efc4/osist-pren-iec-62541-10-2024>

## TABLES

63		
64		
65	Table 1 – Program Finite State Machine .....	4
66	Table 2 – Program states .....	5
67	Table 3 – Program state transitions .....	6
68	Table 4 – Program Control Methods .....	7
69	Table 5 – ProgramStateMachineType .....	10
70	Table 6 – ProgramStateMachineType Attribute values for child Nodes .....	11
71	Table 7 – ProgramStateMachineType Additional References .....	13
72	Table 8 – ProgramStateMachineType causes .....	15
73	Table 9 – ProgramTransitionEventType .....	17
74	Table 10 – AuditProgramTransitionEventType .....	18
75	Table 11 – ProgramDiagnostic2DataType structure .....	19
76	Table 12 – ProgramDiagnostic2DataType definition .....	19
77	Table 13 – ProgramDiagnostic2Type VariableType .....	19
78	Table A.1 – DomainDownload states .....	23
79	Table A.2 – DomainDownloadType .....	25
80	Table A.3 – TransferStateMachineType .....	25
81	Table A.4 – TransferStateMachineType Attribute values for child Nodes .....	26
82	Table A.5 – Finish State Machine Type .....	26
83	Table A.6 – FinishStateMachineType Attribute values for child Nodes .....	27
84	Table A.7 – DomainDownloadType Property Attributes variable values .....	27
85	Table A.8 – TransferStateMachineType Additional References .....	28
86	Table A.9 – Start Method additions .....	30
87	Table A.10 – StartArguments .....	31
88	Table A.11 – IntermediateResults Object .....	32
89	Table A.12 – Intermediate result data Variables .....	33
90	Table A.13 – FinalResultData .....	35
91		
92		

93  
94  
95  
96  
97  
98  
99  
100

# INTERNATIONAL ELECTROTECHNICAL COMMISSION

---

## OPC UNIFIED ARCHITECTURE –

### Part 10: Programs

#### FOREWORD

- 101 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising  
102 all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international  
103 co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and  
104 in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports,  
105 Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their  
106 preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with  
107 may participate in this preparatory work. International, governmental and non-governmental organizations liaising  
108 with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for  
109 Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 110 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international  
111 consensus of opinion on the relevant subjects since each technical committee has representation from all  
112 interested IEC National Committees.
- 113 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National  
114 Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC  
115 Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any  
116 misinterpretation by any end user.
- 117 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications  
118 transparently to the maximum extent possible in their national and regional publications. Any divergence between  
119 any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 120 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity  
121 assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any  
122 services carried out by independent certification bodies.
- 123 6) All users should ensure that they have the latest edition of this publication.
- 124 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and  
125 members of its technical committees and IEC National Committees for any personal injury, property damage or  
126 other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and  
127 expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC  
128 Publications.
- 129 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is  
130 indispensable for the correct application of this publication.
- 131 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent  
132 rights. IEC shall not be held responsible for identifying any or all such patent rights.
- 133 International Standard IEC 62541-10 has been prepared by subcommittee 65E: Devices and  
134 integration in enterprise systems, of IEC technical committee 65: Industrial-process  
135 measurement, control and automation.
- 136 This fourth edition cancels and replaces the third edition published in 2020. This edition  
137 constitutes a technical revision.
- 138 This edition includes the following significant technical changes with respect to the previous  
139 edition:
- 140 a) The StateMachine model which had been in Part 5 is now a separate part – Part 16.  
141 b) StateMachine table format has been aligned.

142 The text of this International Standard is based on the following documents:

CDV	Report on voting
65E/XX/CDV	65E/XX/RVC

143  
144 Full information on the voting for the approval of this International Standard can be found in the  
145 report on voting indicated in the above table.

146 This document has been drafted in accordance with the ISO/IEC Directives, Part 2.

147 Throughout this document and the other parts of the IEC 62541 series, certain document  
148 conventions are used:

149 *Italics* are used to denote a defined term or definition that appears in the "Terms and definition"  
150 clause in one of the parts of the IEC 62541 series.

151 *Italics* are also used to denote the name of a service input or output parameter or the name of  
152 a structure or element of a structure that are usually defined in tables.

153 The *italicized terms and names* are, with a few exceptions, written in camel-case (the practice  
154 of writing compound words or phrases in which the elements are joined without spaces, with  
155 each element's initial letter capitalized within the compound). For example, the defined term is  
156 *AddressSpace* instead of Address Space. This makes it easier to understand that there is a  
157 single definition for *AddressSpace*, not separate definitions for Address and Space.

158 A list of all parts of the IEC 62541 series, published under the general title *OPC Unified*  
159 *Architecture*, can be found on the IEC website.

160 The committee has decided that the contents of this document will remain unchanged until the  
161 stability date indicated on the IEC website under "<http://webstore.iec.ch>" in the data related to  
162 the specific document. At this date, the document will be

- 163 • reconfirmed,
- 164 • withdrawn,
- 165 • replaced by a revised edition, or <http://standards.iso.org/standards/catalog/standards/sist/1b349aca-5812-41e3-88fa-939a42f9efc4/osist-pren-iec-62541-10-2024>
- 166 • amended.

167

**IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.**

168





# OPC Unified Architecture Specification

## Part 10: Programs

### 1 Scope

This part of IEC 62541 defines the *Information Model* associated with *Programs* in OPC Unified Architecture (OPC UA). This includes the description of the *NodeClasses*, standard *Properties*, *Methods* and *Events* and associated behaviour and information for *Programs*.

The complete *AddressSpace* model including all *NodeClasses* and *Attributes* is specified in IEC 62541-3. The *Services* such as those used to invoke the *Methods* used to manage *Programs* are specified in IEC 62541-4.

### 2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments and errata) applies.

IEC 62541-1, *OPC Unified Architecture – Part 1: Overview and Concepts*

IEC 62541-3, *OPC Unified Architecture – Part 3: Address Space Model*

IEC 62541-4, *OPC Unified Architecture – Part 4: Services*

IEC 62541-5, *OPC Unified Architecture – Part 5: Information Model*

IEC 62541-7, *OPC Unified Architecture – Part 7: Profiles*

IEC 62541-16, *OPC Unified Architecture – Part 16: State Machines*

### 3 Terms, definitions and conventions

#### 3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 62541-1, IEC 62541-3, IEC 62541-16, as well as the following apply.

##### 3.1.1

##### Function

programmatic task performed by a *Server* or device, usually accomplished by computer code execution

##### 3.1.2

##### Finite State Machine

sequence of states and valid state transitions along with the causes and effects of those state transitions that define the actions of a *Program* in terms of discrete stages

##### 3.1.3

##### ProgramStateMachineType

type definition of a *Program* and is a subtype of the *FiniteStateMachineType*

##### 3.1.4

##### Program Control Method

*Method* having specific semantics designed for the control of a *Program* by causing a state transition

213 **3.1.5**  
214 **Program Invocation**  
215 unique *Object* instance of a *Program* existing on a *Server*

216 Note 1 to entry: A *Program Invocation* is distinguished from other *Object* instances of the same  
217 *ProgramStateMachineType* by the object node's unique browse path.

### 218 **3.2 Abbreviations**

219 DA Data Access

220 FSM Finite State Machine

221 HMI Human Machine Interfaces

222 UA Unified Architecture

## 223 **4 Concepts**

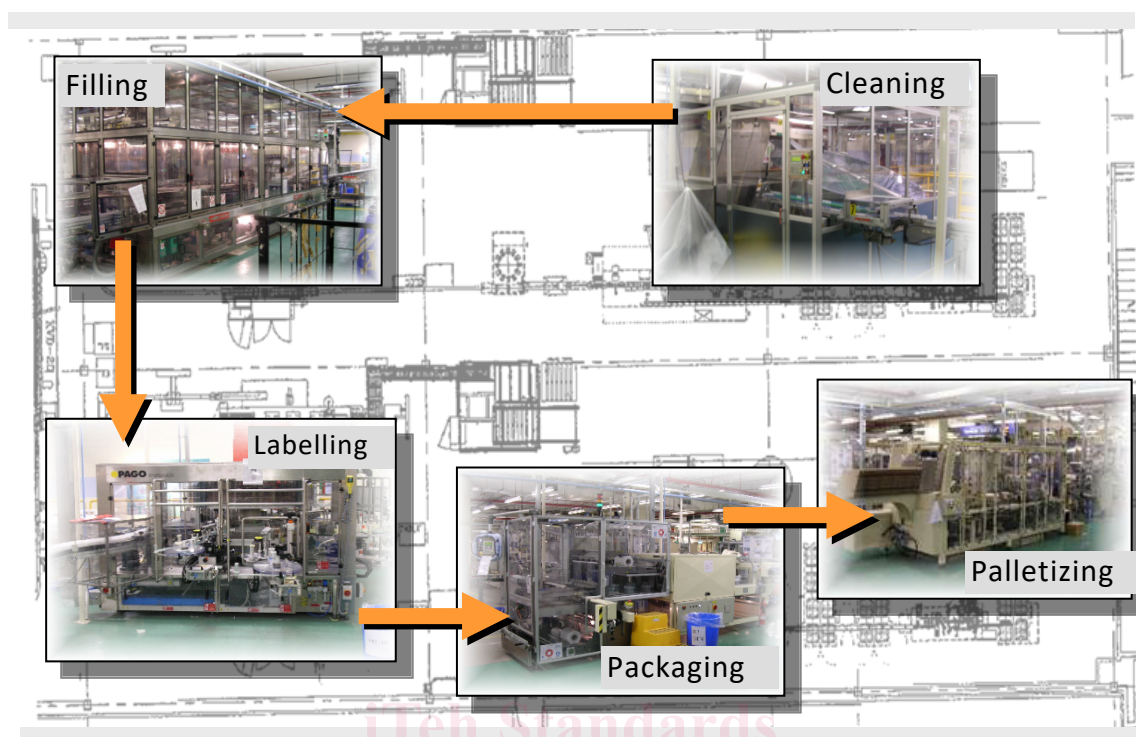
### 224 **4.1 General**

225 Integrated automation facilities manage their operations through the exchange of data and the  
226 coordinated invocation of system *Functions* as illustrated in Figure 1. *Services* are required to  
227 perform the data exchanges and to invoke the *Functions* that constitute system operation.  
228 These *Functions* may be invoked through Human Machine Interfaces, cell controllers, or other  
229 supervisory control and data acquisition type systems. OPC UA defines *Methods* and *Programs*  
230 as an interoperable way to advertise, discover, and request these *Functions*. They provide a  
231 normalizing mechanism for the semantic description, invocation, and result reporting of these  
232 *Functions*. Together *Methods* and *Programs* complement the other OPC UA *Services* and  
233 *ObjectTypes* to facilitate the operation of an automation environment using a client-server  
234 hierarchy.

# iTeh Standards (<https://standards.iteh.ai>) Document Preview

[oSIST prEN IEC 62541-10:2024](https://standards.iteh.ai/catalog/standards/sist/1b349aca-5812-41e3-88fa-939a42f9efc4/osist-pren-iec-62541-10-2024)

<https://standards.iteh.ai/catalog/standards/sist/1b349aca-5812-41e3-88fa-939a42f9efc4/osist-pren-iec-62541-10-2024>



(<https://standards.iteh.ai>)

## Document Preview

Figure 1 – Automation facility control

235

236

237 *Methods* and *Programs* model *Functions* typically have different scopes, behaviours, lifetimes,  
 238 and complexities in *OPC Servers* and the underlying systems. These *Functions* are not normally  
 239 characterized by the reading or writing of data which is accomplished with the *OPC UA Attribute*  
 240 service set.

241 *Methods* represent basic *Functions* in the *Server* that can be invoked by a *Client*. *Programs*, by  
 242 contrast, model more complex and stateful functionality in the system. For example, a method  
 243 call may be used to perform a calculation or reset a counter. A *Program* is used to run and  
 244 control a batch process, execute a machine tool part program, or manage a domain download.  
 245 *Methods* and their invocation mechanism are described in IEC 62541-3 and IEC 62541-4.

246 This standard describes the extensions to, or specific use of, the core capabilities defined in  
 247 IEC 62541-5 and IEC 62541-16 as required for *Programs*.

## 248 4.2 Programs

### 249 4.2.1 Overview

250 *Programs* are complex *Functions* in a *Server* or underlying system that can be invoked and  
 251 managed by a *Client*. *Programs* can represent any level of functionality within a system or  
 252 process in which *Client* control or intervention is required and progress monitoring is desired.  
 253 Figure 2 illustrates the model.